

000007-000008

mom

Please type a plus sign (+) inside this box →

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PTO/SB/05 (4/98)
Approved for use through 09/30/00. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

**UTILITY
PATENT APPLICATION
TRANSMITTAL**

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Atty. Docket No.	X-633 US
------------------	----------

First Inventor or Appl. Identifier	Gi-Joon Nam
------------------------------------	-------------

Title A METHOD AND APPARATUS FOR TESTING ROUTABILITY

Express Mail Label No.	EL539651244US
------------------------	---------------

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Commissioner for Patents
Box Patent Application
Washington, DC 20231

- | | |
|--|--|
| <p>1. <input checked="" type="checkbox"/> *Fee Transmittal Form (e.g., PTO/SB/17)
(Submit an original, and a duplicate for fee processing)</p> <p>2. <input checked="" type="checkbox"/> Specification [Total Pages 18]
(preferred arrangement set forth below)</p> <ul style="list-style-type: none"> - Descriptive title of the Invention - Cross References to Related Applications - Statement Regarding Fed sponsored R & D - Reference to Microfiche Appendix - Background of the Invention - Brief Summary of the Invention - Brief Description of the Drawings (if filed) - Detailed Description - Claim(s) - Abstract of the Disclosure <p>3. <input checked="" type="checkbox"/> Drawing(s) (35 USC 113) [Total Sheets 4]</p> <p>4. Oath or Declaration [Total Pages 3]</p> <ul style="list-style-type: none"> a. <input checked="" type="checkbox"/> Newly executed (original or copy) b. <input type="checkbox"/> Copy from a prior application (37 CFR §1.63(d))
(for continuation/divisional with Box 16 completed) c. <input type="checkbox"/> <u>DELETION OF INVENTOR(S)</u>
Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR § 1.63(d)(2) and 1.33(b). | <p>5. <input type="checkbox"/> Microfiche Computer Program (Appendix)</p> <p>6. <input type="checkbox"/> Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)</p> <ul style="list-style-type: none"> a. <input type="checkbox"/> Computer Readable Copy b. <input type="checkbox"/> Paper Copy (identical to computer copy) c. <input type="checkbox"/> Statement verifying identity of above copies |
|--|--|

ACCOMPANYING APPLICATION PARTS

7. ☒ Assignment Papers (cover sheet & document(s))

8. ☐ 37 CFR §3.73(b) Statement (when there is an assignee) ☐ Power of Attorney

9. ☐ English Translation Document (if applicable)

10. ☐ Information Disclosure Statement (IDS)/PTO-1449 ☐ Copies of IDS Citations

11. ☐ Preliminary Amendment

12. ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)

13. ☐ *Small Entity Statement(s) ☐ Statement filed in prior application, Status still proper and desired (PTO/SB/09-12)

14. ☐ Certified Copy of Priority Document(s)
(If applicable, attach in closed)

NOTE FOR ITEMS 1 & 13: IN ORDER TO BE ENTITLED TO PAY SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R. § 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED UPON (37 C.F.R. § 1.28).

ACCOMPANYING APPLICATION PARTS

7. ☒ Assignment Papers (cover sheet & document(s))
8. ☐ 37 CFR §3.73(b) Statement ☐ Power of Attorney
(when there is an assignee)
9. ☐ English Translation Document (if applicable)
10. ☐ Information Disclosure ☐ Copies of IDS
Statement (IDS)/PTO-1449 Citations
11. ☐ Preliminary Amendment
12. ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
13. ☐ *Small Entity ☐ Statement filed in prior application,
Statement(s) Status still proper and desired
(PTO/SB/09-12)
14. ☐ Certified Copy of Priority Document(s)
(if foreign priority is claimed)
15. ☐ Other:

16. If a **CONTINUING APPLICATION**, check appropriate box and supply the requisite information below and in a preliminary amendment:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: _____
 Prior application information: Examiner _____ Group / Art Unit: _____

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

18. CORRESPONDENCE ADDRESS

☒ Customer Number or Bar Code Label

24309

or ☐ Correspondence address below

(Insert Customer No. or Attach bar code label here)

Name _____

Attn: Edel M. Young

Address

City

State

Zip Code

Country

Telephone

408-879-4969

Fax

408-377-6137

Name (Print/Type)

Edel M. Young

Registration No. (Attorney/Agent)

32,451

Signature

Edel M Young

Date _____

July 25, 2000

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.**

1 A METHOD AND APPARATUS FOR TESTING ROUTABILITY

2 Gi-Joon Nam

3 Sandor S. Kalman

4 Jason H. Anderson

5 Rajeev Jayaraman

6 Sudip K. Nag

7 Jennifer Zhuang

8
9 FIELD OF THE INVENTION

10 The present invention generally relates to analysis of
11 circuit designs, and more particularly to determining the
12 feasibility of routing a plurality of nets on a programmable
13 logic device.

14
15 BACKGROUND

16 The term "net" as used herein refers to an electrical
17 connection between components of a user's circuit design.
18 For example, one net may connect the output terminal of an
19 AND gate to the input terminal of another AND gate and to
20 the input terminal of a flip flop. An AND gate is one
21 component type, and a flip flop is another component type.
22 An "instance" is a single occurrence of a component type. A
23 net connects to a particular pin on a component instance.
24 For example, a clock net is connected to a specific clock
25 pin on a flip-flop component instance. In the context of
26 FPGAs, a pin of a component instance may correspond to the
27 input port of a configurable logic block (CLB), for example.
28 A "netlist" is a list of all the nets which connect the
29 component instances of a user's design and the pins on
30 component instances connected by each net.

31 The circuit design process generally includes the steps
32 of design entry, synthesis, optimization, device mapping,
33 and place-and-route, along with functional and timing
34 simulations to verify correctness. Determining the
35 feasibility of routing a set of nets given certain
36 constraints may be desirable in several situations in the

1 context of this process. Three examples are provided below.
2 In one example involving a given set of nets, it would be
3 useful for a user to know whether there exists a routing
4 solution for the set of nets such that no two nets are
5 electrically shorted in the solution. If no solution
6 exists, the nets may be restructured prior to place-and-
7 route.

8 In another example, a user may need to know the
9 feasibility of routing a set of nets given certain timing
10 constraints. Typically, only a subset of the list of all
11 nets are considered to be "timing-critical" nets. For
12 these timing-critical nets, it would be useful to know
13 whether there exists a routing solution that satisfies the
14 timing constraints. It would be desirable if this
15 determination could be made quickly and prior to routing the
16 set of all nets.

17 In the third example a small change is made to a
18 design. Since the user will not want to disrupt other
19 unchanged and validated sections of the design, it would be
20 useful to determine whether there exists a routing solution
21 for the nets which does not impact the unmodified sections
22 of the design.

23 An apparatus that addresses the aforementioned
24 problems, as well as other related problems, is therefore
25 desirable.

26

27 SUMMARY OF THE INVENTION

28 The present invention generally relates to determining
29 routing feasibility of a plurality of nets. In various
30 embodiments, an associated set of one or more routing
31 solutions are provided for each net, wherein each solution
32 specifies one or more routing resources consumed by the net.
33 A liveness Boolean function is generated having variables
34 that represent respective net/solution pairs. If there
35 exists a set of values for the variables such that at least

one of the variables for each net is logically true, then the liveness function is true. An exclusivity function is generated using the variables that represent the net/solution pairs. If there exists at least one set of values for the variables such that no resource is used by more than a predetermined number of nets, then the exclusivity function is true. The nets are routable using the provided solutions if there exists one set of values for the variables such that both the liveness and exclusivity functions are true.

It will be appreciated that various other embodiments are set forth in the Detailed Description and Claims which follow.

BRIEF DESCRIPTION OF THE DRAWINGS

Various aspects and advantages of the invention will become apparent upon review of the following detailed description and upon reference to the drawings in which:

FIG. 1 is a flowchart of a process for determining whether a set of nets is routable given certain routing resource constraints, according to one embodiment of the invention;

FIG. 2 illustrates an example net table;

FIG. 3 illustrates an example resource table; and

FIG. 4 is a flowchart of a process for generating liveness and exclusivity constraints in accordance with one embodiment of the invention.

DETAILED DESCRIPTION

Boolean Satisfiability (SAT) is a technique that is used to determine whether there exists a satisfying truth assignment for a Boolean function ϕ . Generally, SAT refers to solving a function ϕ that is expressed in conjunctive normal form (CNF), as known to those skilled in the art.

1 In the logical notation used herein, "&" represents a
2 logical AND, "V" represents a logical OR, and "~"
3 represents a logical NOT.

4 In one embodiment, the SAT technique is applied to
5 detailed netlist routing on a programmable logic device
6 (PLD). For example, a field programmable gate array (FPGA)
7 is a PLD having configurable signal-routing resources. The
8 nets in a design are realized physically by assigning
9 routing resources to each net, thereby connecting each net
10 to component instances in the desired manner.

11 FIG. 1 is a flowchart of a process for determining
12 whether a set of nets is routable given certain routing
13 resource constraints, according to one embodiment of the
14 invention. The process generally entails applying the SAT
15 technique to a Boolean function of routability constraints
16 that are generated from input net solutions and the
17 respective routing resources associated with the solutions.

18 At step 102, the nets comprising the circuit design
19 and the solution sets associated with the nets are input
20 for processing. Each net has associated therewith one or
21 more solutions, each solution comprising an associated set
22 of resources. For example, the resources associated with a
23 particular solution may include the routing resources
24 consumed by the solution.

25 In one embodiment, the input nets and solution sets
26 can be input in a text file, in which each net has the
27 following format:

28 **.net** *net_name*
29 **.sol** 1 *resource_1 resource_2 ... resource_i*
30 **.sol** 2 *resource_1 resource_2 ... resource_i*
31 ...
32 **.sol** *m resource_1 resource_2 ... resource_k*

1 .e

2 The keyword ".net" indicates the beginning of a set of
3 routing solutions for the net having the name *net_name*.

4 The keywords ".sol" indicate the beginning of the different
5 routing solutions for the net. Each routing solution

6 includes a solution name, for example, "1" and a list of

7 the routing resources used in the solution, for example,

8 *resource_1 - resource_i*. The ".e" keyword indicates the

9 end of the routing solutions for the net. Each net in the

10 input file has an entry having the format shown above.

11 Those skilled in the art will recognize that the above

12 format is but one example for inputting the routing

13 solutions and that the routing solutions could be input in

14 another form such as a database.

15 At step 104, a net table and a resource table are

16 constructed from the input nets and solution sets. Before

17 continuing with the process of FIG. 1, the net table and

18 resource table are described with reference to FIGs. 2 and

19 3, respectively.

20 FIG. 2 illustrates net table 150. Net table 150

21 includes rows associated with nets 1-r, respectively. Each

22 entry in a row denotes a solution for implementing the net.

23 For example, net 1 has the associated solutions 1-i, and

24 net 2 has the associated solutions 1-j. Recall from the

25 discussion of the information included in the input

26 constraint file that associated with each of the solutions

27 is a set of resources, for example, the routing resources

28 required in the solution.

29 For the purpose of formulating net table 150 into a

30 SAT problem, each of the net/solution pairs is assigned a

31 unique Boolean variable. For example, if net 1 referenced

32 as N1, and solutions 1-i are referenced as S1, S2, S3, and

1 S_i , respectively, then the Boolean variables associated
2 with net 1 are $N1S1$, $N1S2$, $N1S3$, ..., $N1S_i$. Note that the
3 notation $NiSj$ represents a single variable.

4 FIG. 3 illustrates a resource table 180. Resource
5 table 180 is comprised of the set of routing resources used
6 by all of the solutions for all of the nets, along with the
7 net/solution pairs using the associated resources. For
8 example, if nets 1-r (FIG. 2) have solutions that require
9 resources 1-t, the resource table has respective rows for
10 resources 1-t. Each row of resource table 180 provides
11 identification of the net/solution pairs that use the
12 resource associated with the row. One way to identify a
13 net/solution pair is with the associated Boolean variable.
14 Thus, another way to view the rows of the resource table is
15 as lists of Boolean variables that correspond to the
16 net/solution pairs.

17 Returning now to FIG. 1, the net/solution pairs of
18 table 150 and the net/solution pairs in relation to the
19 resources of FIG. 3 are used to construct a Boolean
20 function to which the SAT technique can be applied. At
21 step 106, "liveness" and "exclusivity" constraints are
22 generated using the Boolean variables representing the
23 net/solution pairs and the resources used in the solutions.
24 The liveness and exclusivity constraints ($L(X)$ and $E(X)$)
25 are represented in CNF and are conjoined to form the final
26 routability Boolean function ($R(X) = L(X) \& E(X)$). The
27 routability function constructed from the liveness and
28 exclusivity constraints allows application of the SAT
29 technique to determine whether a layout is impossible given
30 the input net solutions.

Before completing the description of FIG. 1, the generation of the liveness and exclusivity constraints will be discussed with reference to FIG. 4.

FIG. 4 is a flowchart of a process for generating liveness and exclusivity constraints in accordance with one embodiment of the invention. The liveness constraint is constructed to ensure that at least one solution for each of the nets is chosen as a final legal routing solution. The exclusivity constraint is constructed to ensure that no routing resource is used by more nets than it can support.

At step 202, respective Boolean functions are generated for the nets. Each Boolean function is the logical OR of the Boolean variables associated with the solutions that are available for the net. Each solution has a single associated variable. For example, the Boolean function $L(1)$ associated with net 1 of FIG. 2 is written as $L(1) = N1S1 \vee N1S2 \vee N1S3 \vee \dots \vee N1Si$. At step 204, the liveness constraint is generated as a Boolean function $L(X)$ by logically ANDing the Boolean functions $L(1)$, $L(2)$, ..., $L(r)$ associated with the nets. The liveness function is written as $L(X) = L(1) \& L(2) \& \dots \& L(r)$. Note that $L(1)$, $L(2)$, ..., $L(r)$ are referred to as Boolean clauses of the Boolean function $L(X)$. Each clause corresponds to a single net.

Steps 206, 208, and 210 are directed to generating the exclusivity constraint in the form of a Boolean function $E(X)$. Exclusivity is directed to ensuring that a resource is not overused by more than one net. Thus, resources that are used in the solutions of two or more nets are identified at step 206 using resource table 180 (FIG. 3). For each combination including two net/solution pairs that are trying to use a particular resource, for example, net

1 i /solution j and net m /solution n , an exclusivity Boolean
2 clause is constructed to indicate that if net i /solution j
3 is to use the resource, then net m /solution n cannot use
4 the same resource. Specifically, $\sim NiSj \vee \sim NmSn$. Note that
5 exclusivity Boolean clauses are constructed only for
6 combinations of net/solution pairs involving different
7 nets, since a combination of net/solution pairs for the
8 same net would not violate exclusive use of the resource.
9 The exclusivity constraint for a resource r ($E(r)$) is the
10 conjunction of the exclusivity Boolean clauses formed for
11 combinations of net/solution pairs (e.g., $E(r) = (\sim NiSj \vee$
12 $\sim NmSn) \& (\sim NiSj \vee \sim NmSp)$).

13 At step 210, the exclusivity function for the
14 collection of resources ($E(X)$) is formed with a conjunction
15 of the exclusivity constraints of the individual resources.
16 Thus, $E(X) = E(r_1) \& E(r_2) \& E(r_3) \& \dots \& E(r_t)$. The
17 resulting exclusivity function, $E(X)$, is a Boolean function
18 that is in conjunctive normal form (CNF).

19 It will be appreciated that the present invention can
20 be applied to resources other than routing resources. For
21 example, exclusivity constraints could be used to ensure
22 that no two nets route to the same pin. This is applicable
23 in the context of an FPGA since component instances
24 typically have pins that can be swapped, which leads to a
25 situation in which a net routes to any number of pins but
26 no two nets may route to the same pin. In another
27 application, the resources may be the routing channels of
28 an FPGA instead of specific routing resources. In this
29 case, $E(X)$ is generated to ensure that only a limited
30 number of nets are permitted to route in each channel.
31 The logical relationship between the variables in the

1 exclusivity constraints will vary according to the degree
2 to which a resource can be shared between nets.

3 After the liveness and exclusivity constraints have
4 been generated, control is returned to step 108 of FIG. 1,
5 to which this description is redirected.

6 At step 108, the routability function is generated as
7 a conjunction of the liveness and exclusivity constraints
8 ($R(X) = L(X) \& E(X)$). A Boolean satisfiability process is
9 applied to the routability function at step 110 to
10 determine whether the routability function can be
11 satisfied. That is, the Boolean satisfiability process
12 determines whether there is any set of logic values (true
13 or false) for the net/solution variables which would render
14 the routability function logically true. Boolean
15 satisfiability processes are known to those skilled in the
16 art. For example, see J. P. M. Silva and K. A. Sakallah,
17 "GRASP-A New Search Algorithm for Satisfiability," Proc.
18 ACM/IEEE Int. Conference on Computer-Aided Design, Nov.
19 1996, incorporated herein by reference.

20 If the routability function can be satisfied, control
21 is directed to step 112 where the net/solution pairs that
22 solve the routability function are saved. It will be
23 appreciated that the net/solution pairs that solve the
24 routability function are those with the associated Boolean
25 variables having the value TRUE.

26 If two or more solutions have the value TRUE for the
27 same net, only one solution is selected for the final
28 routing. The selection of the solution may be made based
29 on delay and wire length factors associated with the
30 different solutions.

31 Boolean satisfiability typically finds a single
32 assignment of values to variables that satisfies the

1 Boolean function. However, for a particular set of nets,
2 there may exist multiple solution sets which can be found
3 using different, known heuristic algorithms for Boolean
4 satisfiability. If multiple solution sets are identified,
5 overall delay and wire length considerations may be used to
6 select one desirable solution set.

7 If the routability function cannot be satisfied,
8 decision step 114 tests whether the routability process
9 should be repeated with new solution sets. In one
10 embodiment, the routability process continues a selected
11 number of iterations. Alternatively, the routability
12 process can be user controlled and the process stopped at
13 the user's request. In another embodiment, the criteria
14 for stopping the routability process considers the number
15 of solutions provided for each net. For example,
16 initially, a subset of all the available solutions are
17 input to the process. If the routability is not satisfied
18 with the initial set, the number of solutions can be
19 increased for nets that are likely to be impeding the
20 finding of a solution. The process can be repeated until a
21 maximum number of solutions is reached or all possible
22 solutions have been considered.

23 If the routability process is to continue, control is
24 directed to step 116 to determine how to modify the
25 solution sets. One way to modify the solution sets is
26 through an analysis of the liveness function. For the
27 liveness function to evaluate to TRUE, each of its
28 constituent clauses must evaluate to TRUE. Since each
29 liveness clause corresponds to a single net, the number of
30 solutions for a net could be increased if the net's
31 liveness clause could not be satisfied. At step 118, the
32 solution sets are changed. In one embodiment, the results

of the analysis performed at step 116 are used in step 118 to generate additional solutions for selected ones of the nets. In another embodiment, the solution sets of all the nets are modified. Solutions for the nets can be generated automatically using known routing algorithms. For example, see C. Y. Lee, "An Algorithm for Path Connections and its Applications," IRE Transactions on Electronic Computers, Vol. EC=10, 1961, pp. 346-365, and L. McMurchie and C. Ebeling, "PathFinder: A Negotiation-Based Performance-Driven Router for FPGAs," Proc. ACM/SIGDA Int. Symposium on Field Programmable Gate Arrays, 1995, pp. 111-117, both incorporated herein by reference. Once the set of solutions has been changed, processing continues at step 102 to check for routability using the new set of solutions.

The following example illustrates the process of transforming routability constraints into a SAT problem. Consider the following example input file:

```
.net net1
.sol 1 res1 res2 res3
.sol 2 res1 res4 res3
.sol 3 res1 res5 res6 res3
.e

.net net2
.sol 1 res2 res6 res5
.sol 2 res2 res3 res5
.e
```

The input file includes two different nets, net1, and net2, which have three and two different solutions, respectively. Each of the different solutions includes the specified ones of the resources res1-res6.

1 From the input file, the following net table is
2 constructed:

3 net1: net1/sol1 -> net1/sol2 -> net1/sol3
4 net2: net2/sol1 -> net2/sol2

5 The following resource table is also constructed from
6 the input file:

7 res1: net1/sol1 -> net1/sol2 -> net1/sol3
8 res2: net1/sol1 -> net2/sol1 -> net2/sol2
9 res3: net1/sol1 -> net1/sol2 -> net1/sol3 -> net2/sol2
10 res4: net1/sol2
11 res5: net1/sol3 -> net2/sol1 -> net2/sol2
12 res6: net1/sol3 -> net2/sol1

13 A first step of the transformation process is the
14 assignment of a Boolean variable to each net/solution pair,
15 as explained above. It will be appreciated that the number
16 of Boolean variables generated is $O(n)$, where n is the
17 number of net/solution pairs.

18 The liveness function for the input file is:

19 $L(X) = L(\text{net1}) \ \& \ L(\text{net2})$

20 where $L(\text{net1})$ and $L(\text{net2})$ in terms of the Boolean variables
21 are:

22 $L(\text{net1}) = N1S1 \vee N1S2 \vee N1S3$

23 $L(\text{net2}) = N2S1 \vee N2S2$

24 The exclusivity function for the input file is:

25 $E(X) = E(\text{res2}) \ \& \ E(\text{res3}) \ \& \ E(\text{res5}) \ \& \ E(\text{res6})$

26 where $E(\text{res2})$, $E(\text{res3})$, $E(\text{res5})$, and $E(\text{res6})$ in terms of
27 the Boolean variables are:

28 $E(\text{res2}) = (\sim N1S1 \vee \sim N2S1) \ \& \ (\sim N1S1 \vee \sim N2S2)$

29 $E(\text{res3}) = (\sim N1S2 \vee \sim N2S2) \ \& \ (\sim N1S1 \vee \sim N2S2) \ \& \\$
30 $(\sim N1S3 \vee \sim N2S2)$

31 $E(\text{res5}) = (\sim N1S3 \vee \sim N2S1) \ \& \ (\sim N1S3 \vee \sim N2S2)$

32 $E(\text{res6}) = \sim N1S3 \vee \sim N2S1$

1 Since resource res1 is used only in the solutions for net
2 net1, an exclusivity constraint is unnecessary for res1.
3 Similarly, since res4 is used only by the net1/sol2 pair,
4 an exclusivity constraint is unnecessary for res4.

5 The final routability function is $R(X) = L(X) \& E(X)$.
6 Application of the Boolean satisfiability process results
7 in the equation evaluating to true where N1S1 = false, N1S2
8 = true, N1S3 = false, N2S1 = true, and N2S2 = false. In
9 other words, the nets are routable using solution sol2 for
10 net net1 and using solution sol1 for net net2.

11 The present invention is believed to be applicable to
12 a variety of processes for implementing circuit designs and
13 has been found to be particularly applicable and beneficial
14 in PLDs. While the present invention is not so limited, an
15 appreciation of the present invention has been provided by
16 way of specific example nets and routing resources of PLDs.
17 Other aspects and embodiments of the present invention will
18 be apparent to those skilled in the art from consideration
19 of the specification and practice of the invention
20 disclosed herein. It is intended that the specification
21 and illustrated embodiments be considered as examples only,
22 with a true scope and spirit of the invention being
23 indicated by the following claims.

24

CLAIMS

What is claimed is:

1. A computer-implemented method for determining routing feasibility of a plurality of nets, each net having an associated set of one or more routing solutions, each solution using one or more routing resources, comprising:
 - generating a first Boolean function with variables representing respective net/solution pairs, wherein the first function evaluates to true if there exists a set of values for the variables such that at least one of the variables for each net is logically true;
 - generating a second Boolean function using the variables that represent the net/solution pairs, wherein the second function evaluates to true if there exists at least one set of values for the variables such that no resource is used by more than a predetermined number of nets; and
 - outputting a signal indicating whether there exists at least one set of values for the variables for which the first function and the second function evaluate to true.
2. The method of claim 1, further comprising representing the first and second Boolean functions in conjunctive normal form.
3. The method of claim 2, further comprising applying Boolean satisfiability to first and second functions.
4. The method of claim 1, wherein the resources comprise signal routing resources of a field programmable gate array.
5. The method of claim 1, further comprising generating a net table including respective sets of solutions associated with the nets.
6. The method of claim 5, further comprising generating the first function from the net table.

1

2 7. The method of claim 5, further comprising generating a
3 resource table including respective sets of net/solution
4 pairs associated with the resources, wherein each
5 net/solution pair associated with a resource represents
6 usage of the resource by the net/solution pair.

7

8 8. The method of claim 7, further comprising generating
9 the second function from the resource table.

10

11 9. The method of claim 1, wherein the predetermined number
12 of nets that can use a resource is 1.

13

14 10. The method of claim 1, further comprising saving as a
15 routing solution the at least one set of values for the
16 variables for which the first function and the second
17 function evaluate to true.

18

19 11. The method of claim 1, wherein if there exists no set
20 values for the variables for which first and second
21 functions evaluate to true, then performing the steps of:

22 modifying one or more selected sets of the routing
23 solutions; and

24 repeating the steps of generating first and second
25 functions and outputting the signal.

26

27 12. The method of claim 11, further comprising adding
28 additional sets of routing solutions.

29

30 13. The method of claim 12, wherein the repeating step
31 continues until predetermined criteria are met.

32

33 14. An apparatus for determining routing feasibility of a
34 plurality of nets, each net having an associated set of one
35 or more routing solutions, each solution using one or more
36 routing resources, comprising:

1 means for generating a first Boolean function with
2 variables representing respective net/solution pairs,
3 wherein the first function evaluates to true if there exists
4 a set of values for the variables such that at least one of
5 the variables for each net is logically true;

6 means for generating a second Boolean function using
7 the variables that represent the net/solution pairs, wherein
8 the second function evaluates to true if there exists at
9 least one set of values for the variables such that no
10 resource is used is by more than a predetermined number of
11 nets; and

12 means for outputting a signal indicating whether there
13 exists at least one set of values for the variables for
14 which the first function and the second function evaluate to
15 true.

16
17 15. A computer-implemented method for determining routing
18 feasibility of a plurality of nets sharing a plurality of
19 resources, comprising:

20 inputting respective sets of one or more solutions
21 associated with the plurality of nets, each solution being
22 associated with one or more required resources
23 assigning respective identifiers to net/solution pairs;
24 generating respective Boolean liveness functions for
25 the nets using the net/solution pairs;

26 generating respective Boolean exclusivity functions
27 using the net/solution pairs for each resource required by
28 two or more nets;

29 generating a routability Boolean function as a logical
30 AND of the liveness functions and exclusivity functions; and

31 testing whether the routability function can be
32 satisfied.

33
34 16. The method of claim 15, further comprising generating a
35 net table including respective sets of solutions associated
36 with the nets.

1

2 17. The method of claim 16, further comprising generating
3 the liveness functions from the net table.

4

5 18. The method of claim 15, further comprising generating a
6 resource table including respective sets of net/solution
7 pairs associated with the resources, wherein each
8 net/solution pair associated with a resource represents
9 usage of the resource by the net/solution pair.

10

11 19. The method of claim 18, further comprising generating
12 the exclusivity functions from the resource table.

13

14 20. The method of claim 15, further comprising representing
15 the liveness and exclusivity functions and the routability
16 function in conjunctive normal form.

17

18 21. The method of claim 14 further comprising applying
19 Boolean satisfiability to the routability function.

20

1
2 A METHOD AND APPARATUS FOR TESTING ROUTABILITY

3 Gi-Joon Nam

4 Sandor S. Kalman

5 Jason H. Anderson

6 Rajeev Jayaraman

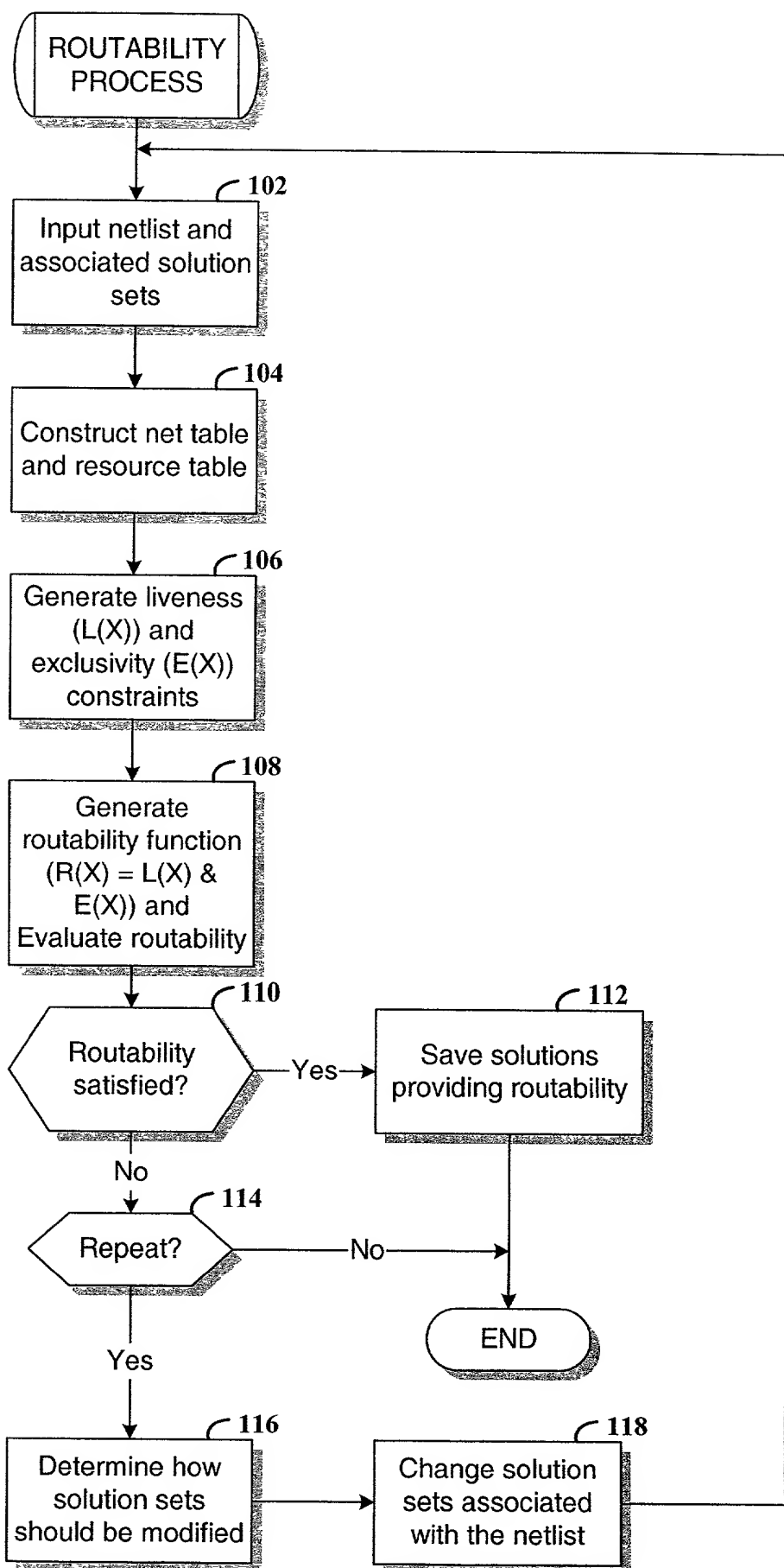
7 Sudip K. Nag

8 Jennifer Zhuang

9
10
11 ABSTRACT

12 A method and apparatus for determining routing
13 feasibility of a plurality of nets. Each net has an
14 associated set of one or more routing solutions, wherein
15 each solution specifies one or more routing resources
16 consumed by the net. A liveness Boolean function is
17 generated having variables that represent respective
18 net/solution pairs. If there exists a set of values for the
19 variables such that at least one of the variables for each
20 net is logically true, then the liveness function is true.
21 An exclusivity function is generated using the variables
22 that represent the net/solution pairs. If there exists at
23 least one set of values for the variables such that no
24 resource is used is by more than a predetermined number of
25 nets, then the exclusivity function is true. The nets are
26 routable using the provided solutions if there is one set of
27 values for the variables such that both the liveness and
28 exclusivity functions are true.

FIG. 1



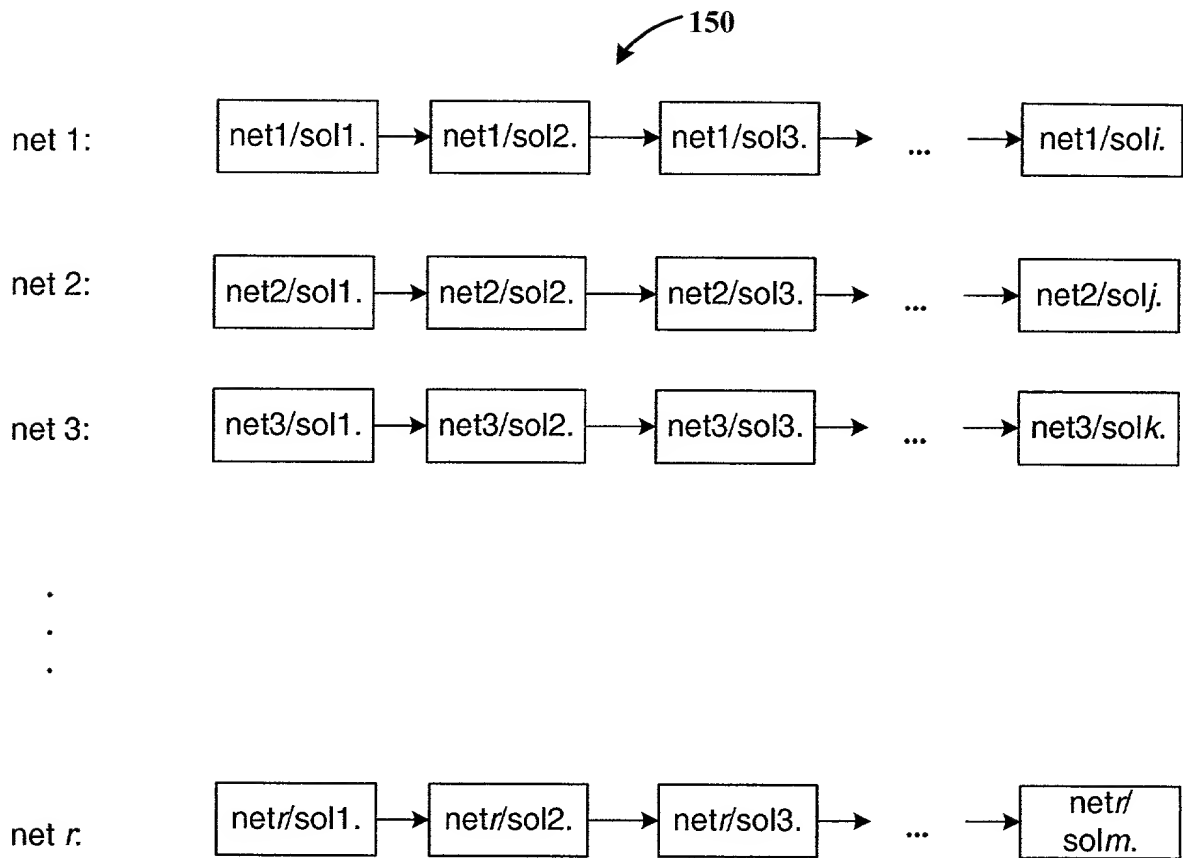


FIG. 2

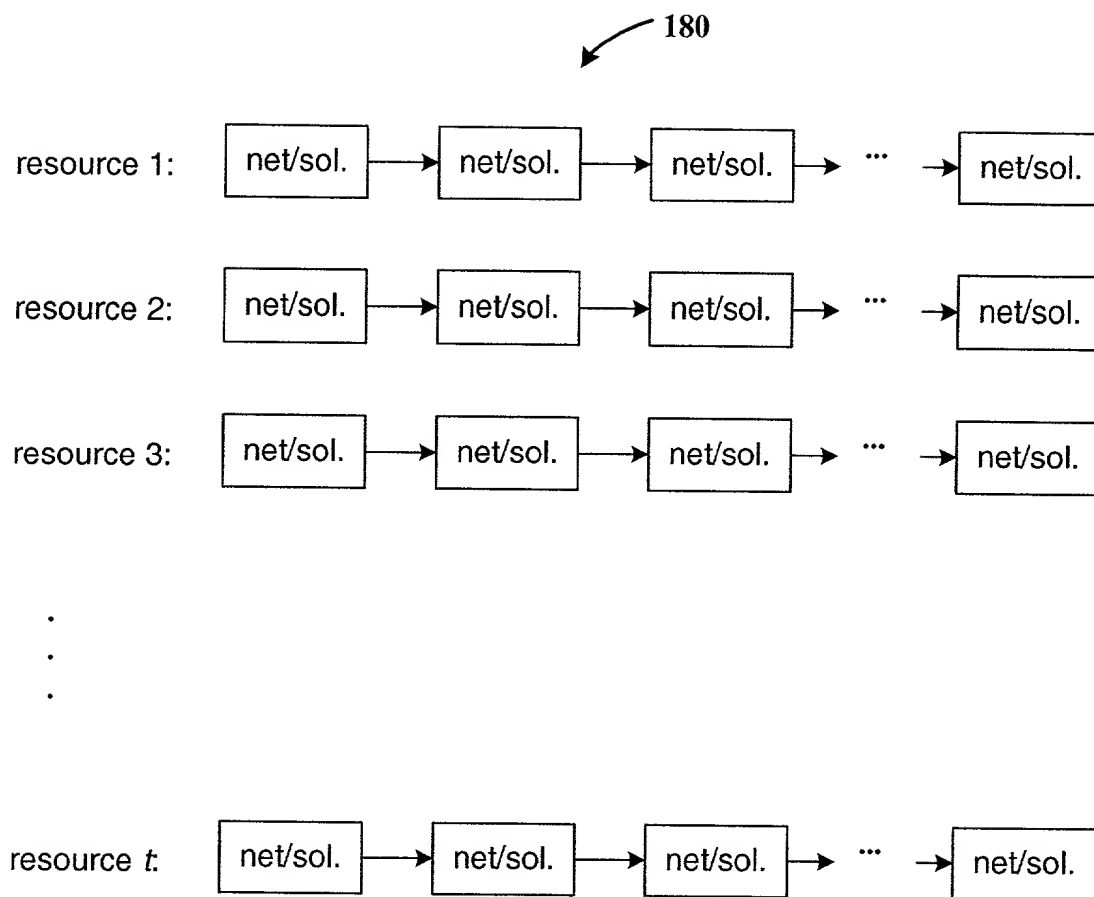
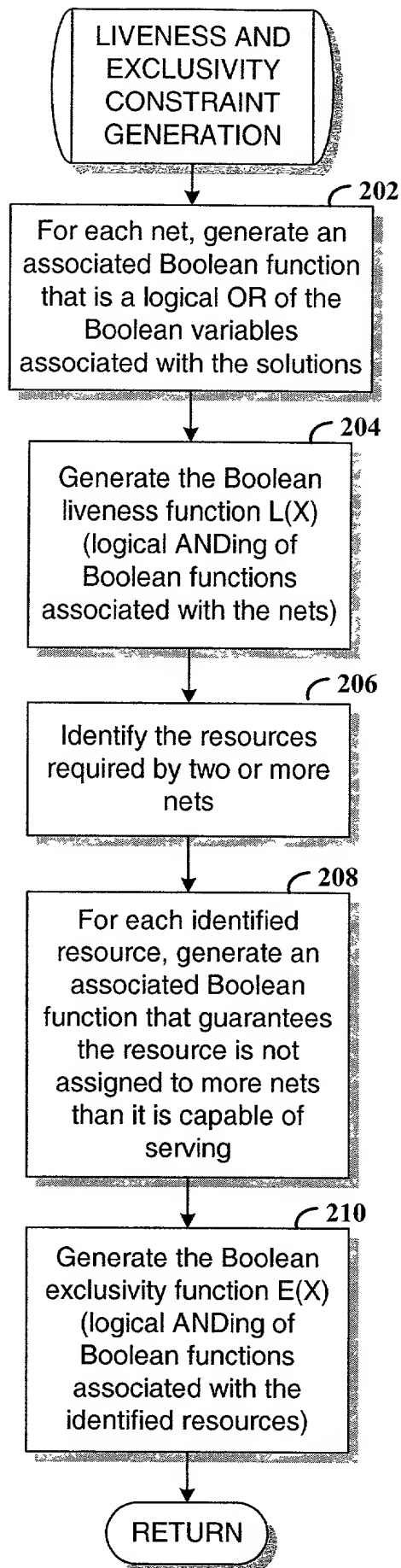


FIG. 3

**FIG. 4**

DECLARATION FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below adjacent to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of subject matter (process, machine, manufacture, or composition of matter, or an improvement thereof) that is disclosed and/or claimed and for which a patent is solicited by way of the application entitled

A METHOD AND APPARATUS FOR TESTING ROUTABILITY

which (check)

- ☒ is attached hereto.
☐ and is amended by the Preliminary Amendment attached hereto.
☐ was filed on _____ as Application Serial No. _____.
☐ and was amended on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified application, including the claims, including portions amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, § 1.56(a).

I hereby claim foreign priority benefits under title 35, United States Code, § 119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
N / A			Yes	No
(Number)	(Country)	(Day/Month/Year Filed)		
(Number)	(Country)	(Day/Month/Year Filed)	Yes	No

I hereby claim the benefit under 35 U.S.C. § 119(e) of any United States provisional application(s) listed below

N / A
(Application Number(s)) (Filing Date (MM/DD/YYYY))

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States application(s) listed below and, insofar as any subject matter of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, § 1.56(a) which occurred between the filing date of the prior application(s) and the national or PCT international filing date of this application:

N / A
(Application Serial No.) (Filing Date) (Status-patented, pending, abandoned)
(Application Serial No.) (Filing Date) (Status-patented, pending, abandoned)

I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and to transact all business in the United States Patent and Trademark Office connected herewith:

Keith A. Chanroo (36,480), Edel M. Young (32,451), and Lois D. Cartier (40,941) all located at Xilinx, Inc., 2100 Logic Drive, San Jose, California 95124, and

LeRoy D. Maunu (35,274) located at Crawford, PLLC, 333 Washington Avenue N, Suite 5000, Minneapolis, Minnesota 55401.

Address all telephone calls to: Edel M. Young at Telephone No. (408) 879-4969

Address all correspondence to: Edel M. Young
Xilinx, Inc.
2100 Logic Drive
San Jose, California 95124

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Title 18, United States Code, § 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of sole or first inventor Gi-Joon Nam

Inventor's signature *Gi-Joon Nam* Date July 18, 2000

Residence Ann Arbor, Michigan 48105 Citizenship Korea

Post Office Address 3566 Green Brier Blvd., #411C, Ann Arbor, Michigan 48105

Full name second inventor Sandor S. Kalman

Inventor's signature *Sandor Kalman* Date July 25, 2000

Residence Los Gatos, California 95030 Citizenship United States

Post Office Address ~~127 El Olivar, Los Gatos, California 95030~~
734 HILMAR ST. SANTA CLARA, CALIFORNIA 95050

Full name third inventor Jason H. Anderson

Inventor's signature *Jason H. Anderson* Date July 21, 2000

Residence Campbell, California 95008 Citizenship Canada

Post Office Address ~~325 Union Avenue, Apt. 355, Campbell, California 95008~~
255 Gomes Ct. #3, Campbell, CA 95008

Full name fourth inventor Rajeev Jayaraman

Inventor's signature *Rajeev Jayaraman* Date July 21, 2000

Residence San Jose, California 95131 Citizenship India

Post Office Address 1859 Tersini Court, San Jose, California 95131

EMY:pbs

DECLARATION FOR PATENT APPLICATION
(Additional Inventors)Full name of fifth inventor Sudip K. NagInventor's signature *Sudip K. Nag* Date July 21, 2000Residence San Jose, California 95132 Citizenship IndiaPost Office Address 345 Kiely Boulevard, #C212, San Jose, California 95132Full name sixth inventor Jennifer ZhuangInventor's signature *Jen Zhuang* Date July 21, 2000Residence San Jose, California 95129 Citizenship ~~P.R. China~~ USAPost Office Address 1640 Petal Way, San Jose, California 95129

Full name seventh inventor _____

Inventor's signature _____ Date _____

Residence _____ Citizenship _____

Post Office Address _____

Full name eighth inventor _____

Inventor's signature _____ Date _____

Residence _____ Citizenship _____

Post Office Address _____

Full name ninth inventor _____

Inventor's signature _____ Date _____

Residence _____ Citizenship _____

Post Office Address _____

Full name tenth inventor _____

Inventor's signature _____ Date _____

Residence _____ Citizenship _____

Post Office Address _____